Fermilab

FN-287
0200

MESSYMESH - A COMPUTER PROGRAM TO CALCULATE LINEAR ACCELERATOR CAVITY FIELDS

D. E. Johnson

December 10, 1975

Introduction

MESSYMESH, an overrelaxation program designed to calculate electromagnetic eigenfunctions and eigenvalues associated with an Alvarez type linear accelerator cavity, was developed at MURA to run on an IBM 704 computer.[1-3] This program, along with associated output manipulating programs, has been extensively modified over the years and now exists as a working program in standard ASCII FORTRAN on a CDC 6600 computer at Fermilab. The program requires one nine-track tape for primary output and takes approximately five minutes of run time to originally compute the eigenvectors. It occupies 112k octal words of fast memory. Subsequent iterations, starting from a previously created mesh stored on magnetic tape, require run times of between 30 seconds and one minute.

MESSYMESH, along with the associated programs TPPREP and BCDSUM, compute and print out eigenfunctions and eigenvalues, and several other quantities of interest, such as transit time factors, stored energy, shunt impedance, etc., for linear accelerator cavities loaded with cylindrical drift tubes with either square or circular corners.

This report describes briefly the method of calculation used by MESSYMESH and also gives procedural instructions in the use of the programs. Much of a previous MURA report[2] has been reproduced here, either in original or necessarily modified form, for the sake of completeness.

Equations[2]

A detailed analysis of the linear accelerator problem, with emphasis on the variational principle associated with the solution has previously been given.[1] Hence only pertinent final formulae employed by MESSYMESH are reproduced here.

Maxwell's electromagnetic field equations yield, for azimuthally independent modes, a scalar wave equation for the azimuthal component of the magnetic field:

$$\frac{\partial^2 F}{\partial r^2} - \frac{1}{r}\frac{\partial F}{\partial r} + \frac{\partial^2 F}{\partial Z^2} + k^2 F = 0 \ , \tag{1}$$

$$\frac{\partial F}{\partial n} = 0 \quad \text{on conductor}$$

where $F = rH_\phi$ and where the boundary condition is to be applied on all metallic and nodal surfaces as shown in Fig. 1 and where in addition $F(0,Z) = 0$.

The above differential equation is solved for the geometry shown in Fig. 1 by establishing a mesh or grid in the region penetrated by the field, in a way such that a mesh line coincides everywhere with the metallic and nodal boundaries. Obviously, this would be very difficult to accomplish in the most general case; therefore, the geometries for which MESSYMESH will provide answers are limited to cylindrical walled cavities with azimuthally symmetric drift tubes with or without holes and with circularly bounded or square outer corners and hole corners. In order to achieve solutions for these restricted geometries, overlapping rectangular and circular polar meshes are established, such that the boundary everywhere does coincide with a mesh line.

Transforming to polar coordinates

$$r = r_x + \rho \sin\theta$$

$$Z = Z_c + \rho \cos\theta$$

with r_c and Z_c being the r and Z rectangular coordinates of the center of curvature, equation (1) becomes

$$\frac{\partial^2 F}{\partial \rho^2} + \frac{r_c}{\rho(r_c+\rho \sin\theta)} \frac{\partial F}{\partial \rho} + \frac{1}{\rho^2} \frac{\partial^2 F}{\partial \theta^2} - \frac{\cos\theta}{\rho(r_c+\rho \sin\theta)} \frac{\partial F}{\partial \theta} + k^2 F = 0. \qquad (2)$$

Finite difference approximations to the differential equation may be found for the square mesh and polar meshes. The Liebmann four point algorisms for the two mesh types are, respectively:

$$F^{n+1}_{i,j} - F^n_{i,j} = \alpha \left\{ F^n_{i,j} \left(\frac{k^2 h^2}{4} - 1 \right) + \frac{1}{4} \left[F^n_{i,j+1} \left(1 - \frac{h}{2r} \right) + F^{n+1}_{i,j-1} \left(1 + \frac{h}{2r} \right) + \right. \right.$$

$$\left. \left. + F^n_{i+1,j} + F^{n+1}_{i-1,j} \right] \right\} ; \qquad (3)$$

$$F^{n+1}_{\ell,m} - F^n_{\ell,m} = \alpha \left\{ F^n_{\ell,m} \left[\frac{k^2}{\frac{2}{h_\rho^2} + \frac{2}{h_\theta^2}} - 1 \right] + \right.$$

$$+ \frac{1}{\left(\frac{2}{h_\rho^2} + \frac{2}{h_\theta^2} \right)} \left[F^n_{\ell+1,m} \left\{ \frac{1}{h_\rho^2} + \frac{1}{2h_\rho} \left[\frac{r_c}{\rho(r_c+\rho \sin\theta)} \right] \right\} + \right.$$

$$+ F^{n+1}_{\ell-1,m} \left[\frac{1}{h_\rho^2} - \frac{1}{2h_\rho} \left[\frac{r_c}{\rho(r_c+\rho \sin\theta} \right] \right] +$$

$$+ F^{n}_{\ell,m+1}\left\{\frac{1}{h_\theta^2} - \frac{1}{2h_\theta}\left[\frac{\cos\theta}{r_c+\rho\sin\theta}\right]\right\} +$$

$$+ F^{n+1}_{\ell,m-1}\left\{\frac{1}{h_\theta^2} + \frac{1}{2h_\theta}\left[\frac{\cos\theta}{r_c+\rho\sin\theta}\right]\right\}\Bigg]\Bigg\} \qquad (4)$$

where h, h_θ, and h_ρ are rectangular and polar mesh spacings, respectively, n and n+1 refer to the iteration number, and i,j and ℓ,m refer to the rectangular and polar mesh indexing respectively. Repetitive calculation of improved field values at the mesh points leads to an iterative convergence, provided that α, the over-relaxation factor is properly (and perhaps optimally) chosen.

Simultaneous to the improvement of each of the mesh point field values from an initially guessed set of values to the correct eigenfunction, a calculation of the eigenvalue must take place. A variational principle indicates that a lower limit exists for the calculation:

$$K^2 = - \frac{\langle F,\mathfrak{L}F\rangle}{\langle F,F\rangle} = - \frac{\iint_s \frac{F}{r}\left[\frac{\partial^2 F}{\partial r^2} - \frac{1}{r}\frac{\partial F}{\partial r} + \frac{\partial^2 F}{\partial Z^2}\right] drdZ}{\iint_s \frac{F^2}{r}\, drdZ} \qquad (5)$$

The lower limit of this calculation occurs when F is the eigenfunction of the differential equation. In this event K is just the corresponding eigenvalue k. (In the presence of non-rectangular boundaries, the variational principle is apparently only approximately true.)

Calculation of improved eigenvalues can be reduced to finite difference approximations similar to the algorisms given above. For the square and polar meshes the expressions become, respectively:

$$\frac{k^2h^2}{4} = \frac{\sum\limits_i \sum\limits_j F^n_{i,j} \; A\dot{\Delta} S_{i,j}/r}{\sum\limits_i \sum\limits_j \frac{\left(F_{i,j}\right)^2}{r_j} \Delta S_{i,j}}$$

with

$$A = \frac{1}{4}\left[F^n_{i,j+1}\left(1-\frac{h}{2r_j}\right) + F^n_{i,j-1}\left(1+\frac{h}{2r_j}\right) + F^n_{i+1,j} + F^n_{i-1,j}\right] - F^n_{i,j} \tag{6}$$

and $$\frac{k^2h^2}{4} = \frac{B}{C} \tag{7}$$

where

$$B = \sum_\ell \sum_m \left\{ \left[\frac{F^n_{\ell,m}}{\frac{r_c}{h} + \frac{\rho \sin\theta}{h}} \right] * \frac{1}{4}\left[\frac{h^2}{h_\rho{}^2}\left(F^n_{\ell+1,m} + F^n_{\ell-1,m}\right) + \frac{h^2}{h_\theta{}^2}\left(F^n_{\ell,m+1} + F^n_{\ell,m-1}\right) \right.\right.$$

$$+ \left(\frac{h}{h\rho}\right) \frac{F^n_{\ell+1,m} - F^n_{\ell-1,m}}{2\frac{\rho}{r_c}\left(\frac{r_c}{h} + \frac{\rho \sin\theta}{h}\right)} + \left(\frac{h}{h_\theta}\right) \frac{\cos\theta \left(F^n_{\ell,m-1} - F^n_{\ell,m+1}\right)}{2\left(\frac{r_c}{h} + \frac{\rho \sin\theta}{h}\right)} +$$

$$\left.\left. - 2\left(\frac{h^2}{h_\rho{}^2} + \frac{h^2}{h_\theta{}^2}\right) F^n_{\ell,m} \right] \right\} \Delta S_{\ell,m}$$

and

$$C = \sum_\ell \sum_m \frac{\left(F^n_{\ell,m}\right)^2}{\left(\frac{r_c}{h} + \frac{\rho \sin\theta}{h}\right)} \Delta S_{\ell,m} \; .$$

Alternative improvement of the field values and calculation of an improved eigenvalue is thus the method of solution.

Because of the existance of the variational principle, a convergent process is virtually assured, provided α, the over-relaxation parameter, is properly chosen.

Employment of overlapping meshes necessitates interpolation of field values, etc. between meshes. Consistency with the differential equation and the accuracy of the above algorisms, etc. requires that a field value on one mesh be given in terms of five field values on the other overlapping mesh such that, for example:

$$F_{i,j} = \frac{\lambda_1 F_{00} + \lambda_2 F_{01} + \lambda_3 F_{10} + \lambda_4 F_{11} + \lambda_5 F_{ab}}{\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 - k^2}$$

where the λ's are determined via a 5^{th} order matrix solution previously specified.

Formulae for the auxiliary quantities calculated by MESSYMESH include the following.(1)

Energy Stored Per Unit Volume: $\frac{W}{V} = \frac{u}{2} \frac{\iint \frac{F^2}{r}\, dS}{\iint r\, dS}$

Power Lost to Cavity Walls: $P_W = \pi R_S \oint \frac{F^2}{r}\, d\ell$

Q Factor: $Q = \frac{\omega u}{R_s} \frac{\iint \frac{F^2}{r}\, dS}{\oint \frac{F^2}{r}\, d\ell}$

Average Axial Accelerating Field: $E_o = \frac{1}{\omega\varepsilon L} \int_{-L/2}^{L/2} \left(\frac{1}{r}\frac{\partial F}{\partial r}\right)_{0,Z} dZ$

Shunt Impedance: $Z_S = E_o^{\,2}/(Pw/L)$

Transit Time Factor: $T = \frac{1}{\omega\varepsilon L E_0} \int_{-L/2}^{L/2} \left(\frac{1}{r}\frac{\partial F}{\partial r}\right)_{0,Z} \frac{\cos\, 2\pi Z}{L}\, dZ$

Coupling Coefficient: $S = \frac{1}{\omega\varepsilon L^2\, E_0} \int_{-L/2}^{L/2} \left(\frac{1}{r}\frac{\partial F}{\partial r}\right)_{0,Z} Z \sin \frac{2\pi Z}{L}\, d$

These quantities are calculated from the final converged fields and, together with the eigenvalue, serve as indicators of the geometry for which they have been calculated.

Use of Programs

TPPREP

In order to use the program MESSYMESH, one must first prepare a tape in a manner acceptable to the program. This tape is used for output from a MESSYMESH run, which consists of geometrical data and auxiliary output quantities, and the final mesh field values for each of the various types of meshes - square, curvilinear drift tube corner, and curvilinear hole corner. Each run produces one such set of output, and 100 sets may be stored on one tape. The tape must be prepared, via the program TPPREP, with a label identifying it as a MESSYMESH tape and with a specified range of allowable iteration numbers, each MESSYMESH run being assigned a unique number. The use of TPPREP is illustrated in Appendix A.

MESSYMESH Calculation

Having prepared a mesh tape, one may then proceed to calculate the field quantities associated with a particular cavity geometry. The geometrical data must be scaled such that the drift tube dimensions are integers, and thus, an integer number of mesh units. This is illustrated in Appendix B.

Initially, the eigenvalues are calculated from a slightly modified bessel function load, and the program proceeds, through an overrelaxation method, until two input convergence criteria are satisfied. This takes typically five minutes on a 6600. At the end of the run, the final mesh, along with the labeling iteration number and the pertinent geometrical data, are stored on the previously prepared output tape mounted on logical unit 20.

Subsequent runs may then be computed starting with a previously calculated final mesh as initial load. This may be done in order to improve the convergence of a particular run through re-iteration. Also, a different cavity geometry may be calculated using an old mesh, as the program will check the mesh with the geometrical data and automatically expand or contract it in order to fit the desired cavity. Thus, one may change the length of a drift tube, for example, and quickly calculate a new cavity without having to revert back to a bessel function load.

BCDSUM

The program BCDSUM is used to obtain useful, non-scaled answers from the MESSYMESH mesh tape. One inputs a scale factor, type of drift tube, and average axial electrical field desired, along with a mesh number which has been previously calculated and stored on the output tape, and the program produces a one page listing of the quantities of interest for a linac cavity. The use of BCDSUM and a typical output page are shown in Appendix C.

References

1. T.W. Edwards, "Proton Linear Accelerator Cavity Calculations", MURA report #622, 1961

2. T.W. Edwards, "MESSYMESH", MURA report #642, 1962

3. B. Austin, et al, "The Design of Proton Linear Accelerators for Energies Up to 200 MeV", MURA report #713

FN-287
0200

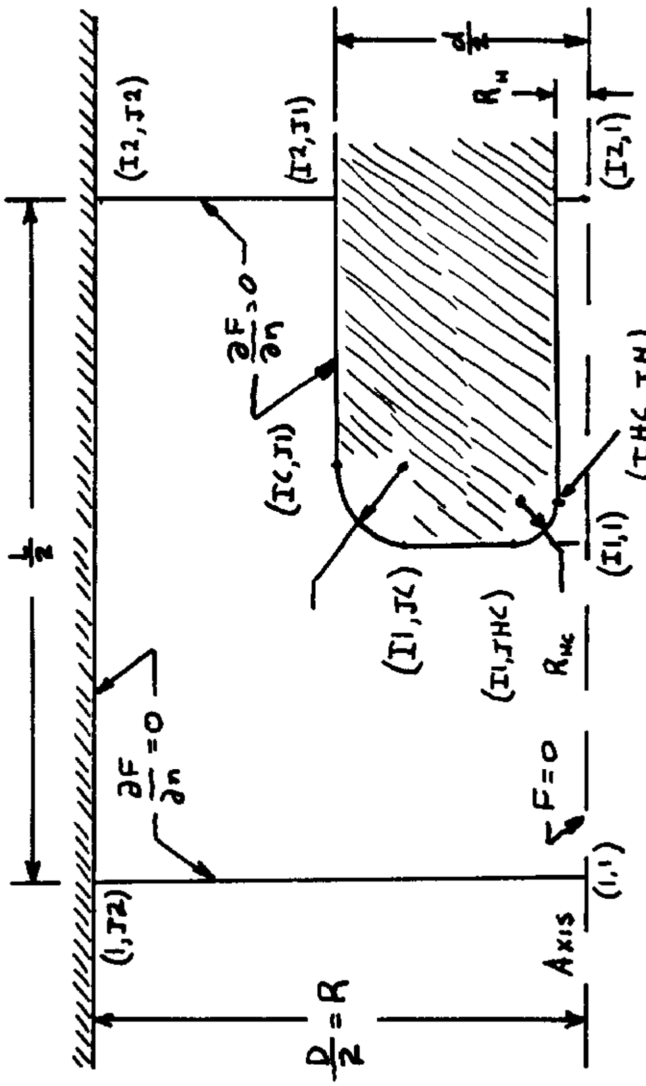

Fig. 1. One fourth of a meridian plane crossection of a linear accelerator cavity.

Appendix A

TPPREP - Mesh Tape Preparation Program

This program is designed to put a unique three word record followed by an end-of-file on MESSYMESH mesh tapes. The format of this record is:

WORD 1: Identification Word

WORD 2: Lowest Allowed Mesh Number

WORD 3: Highest Allowed Mesh Number.

The program requires a nine-track tape mounted on logical unit 20, and, if desired, a second tape, also to be prepared, mounted on logical unit 19. Data is read in following a FORTRAN (I3, E15.7) format, and consists of:

Data

Column 3	Column 18	
1	10000.0	lowest mesh number allowable
2	10100.0	highest mesh number allowable
3	1.0	flag for preparing two tapes (= 0.0 for 1 tape)

Blank Card

1	10101.0	lowest mesh number on second tape
2	10200.0	highest mesh number on second tape

Two Blank Cards

Remarks

1. Data sets must be separated by one blank card and terminated with two blank cards.
2. After the unique first record is put on the tape, an end-of-file is also written. When a mesh is written on this tape, the end-of-file is erased and re-written following the mesh.

3. This program will not prepare (anew) a tape that is already a MESSYMESH mesh tape. An on-line diagnostic will be printed accordingly to instruct the operator.

Appendix B

MESSYMESH - Main Calculating Program

This program is designed to calculate eigenfunctions, eigenvalues, and auxiliary quantites for linear accelerator cavities of the type depicted in Fig. 1. This program requires the previously prepared tape mounted on logical unit 20. Data is input via a FORTRAN (I3,E15.7) format, and consists of:

Column 3	Column 18	
1	10002.0	identification # of mesh being calculated
2	10001.0	mesh # of initial load source
		> 0 loads from tape 20 or 21.
		= 0 loads from Bessel function
		< 0 loads from core previous run
3	2.0	profile identification #
		= 1 square corner drift tube
		= 2 round corner drift tube
4	16.597	scaled size of g/2. Must be ≥2.
5	48.698	scaled size of L/2. Must be ≥ g/2.
6	16.0	scaled size of d/2. Must be ≥2.
7	88.05	scaled size of D/2. Must be ≥ d/2+2.
8	3.0	scaled size of Rh
9	2.0	scaled size of Rh_c
10	8.0	scaled size of R_c

Comments:

1. d/2, rh, rh_c, and r_c must be integers when scaled
2. For all runs (L/2 scaled) x (D/2 scaled) must be ≤10,000 and L/2 scaled must be ≤200, D/2 scaled must be ≤500.
3. For an unloaded cavity address 6 (d/2) must be zero.

Appendix B
(Cont.)

Column 3

12	1.75	Initial value of alpha, the over relaxation factor. If not specified the program assumes minimum of tape loaded and 1.7.
13	9.5-7	Normalization value of stored energy. If not specified, program assume $W = 9.791 \times 10^{-7}/R^2$ joules/cubic meter.
14	5.0-4	First convergence criterion (change in eigenvalue per iteration). If not specified, program assumes PC = 10-3.
15	1.0-11	Second convergence criterion (average residual per mesh point). If not specified, program assumes SCC = 10^{-10}.
20	1.0	De-bug prints from SETUP (none if = 0)
21	1.0	De-bug prints from EXPAND (none if = 0
22	1.0	De-bug prints from ITERAT (none if = 0)
23	1.0	De-bug prints from ALG (none if = 0)
24	1.0	De-bug prints from ALGC (none if = 0)
35	1.0	Print out calculated convergence data after each iteration.
36	≠0.0	Print out final F values
37	≠0.0	Print out scaled linac quantities periodically

COMMENTS

1. Only data address 1 must be changed for each run. Addresses 2 thorugh 15 will remain unchanged if not specified differently. Addresses 20 through 24 are set to zero between runs to reduce the amount of off line output.

2. Input for successive runs must be separated by one blank card and the last run must be followed by two blank cards.

3. Each run must be assigned a unique number. The program will not write two meshes with the same number on an input/output mesh tape.

4. Meshes may be loaded from or stored on mesh input/output tapes logical numbers 20 or 21. Tape 20 is always required. The use of tape 21 is optional. If tape 21 is used tape 20 will be interrogated first, both before loading or storing meshes via tape 21. If tape 20 is the correct one, 21 will not be used.

5. The meshes stored on Mesh input/output tapes must be preceded by a unique three word record, written by TPPREP. This record determines which (ident #) meshes may be stored on that particular tape.

6. If an identification number is specified for a mesh which cannot be stored on either tape 20 or 21 the program will come to a stop.

7. The program analyzes the input data to see if it is geometrically sensible. It it is not, the run is aborted and the next one attempted.

8. If a run is aborted as above the next run will not be possible if a core load is required. In this event, the succeeding run will be attempted, etc.

9. The format of each mesh stored on a mesh input/output is as follows:

Record 1: 40 words containing geometrical data and output auxiliary quantities.

Record 2 through I2+1: Square mesh final field values. Each record contains J2+1 words (first word blank). The mesh field values are stored column-wise beginning at the lower left corner.

Next 3 records: Curvilinear drift tube corner mesh final field values, if such exist. Existance is indicated in Record 1.

Next 3 records: Curvilinear hole corner mesh final field values, if such exist. Existance is indicated in Record 1.

10. An End-of-File gap is always searched for, erased, and after writing a mesh on an input/output tape, restored. This End-of-File signifies the end of useful information on an input/output tape.

11. Use of data addresses 20 through 24 and 35 through 37 insures voluminous off-line output, especially address 23 and 24. Extreme discretion should be employed in using these. It is doubtful that an entire mesh calculation, when in this mode, is practical because of the amount of output.

12. The rate of convergence is determined by the quality alpha (data address 12), the over relaxation parameter. Periodic calculation of the optimum/maximum value takes place as the iterative process proceeds. This parameter is quite critical. If it is too large divergence may occur (the program corrects for this). It if is too small, convergence will be very slow. Judicious care should be used in manipulating this parameter.

Subroutines and Their Functions

The hand coded program and subroutines can be divided into several categories on the basis of their purpose. Below is a brief indication of their purpose(s).

Main Control
Program: Exercises control over loading, initiating, printing, iteration, interpolating, aborting and saving, normalizing, and outputting processes

MURCD2: Reads in data cards.

Loading subroutines

INPUT: Reads data cards via MURCD2, searches input tape(s) for required loading mesh.

Initiating Subroutines

SETUP: Converts input data to useable form, controls establishment of interpolation coefficients, etc.

ALGCOF: Calculates interpolation coefficients.

SETSPC: Calculates backward interpolation coefficients.

SETCOF: Calculates forward interpolation coefficients.

PACK: Packs forward and backward interpolation coefficients.

MATIN1: Matrix inversion subroutine.

EXPAND: Modifies mesh dimensions from those of mesh loaded from tape to those of run to be calculated. Interpolates field values accordingdly.

Printing Subroutines

ERRPRT: Prints out input data error diagnoses.

PRTHED: Prints out page heading at beginning of each run.

PRTFLD: Prints out fields when requested and at end of program.

COLHED: Prints out column heading for PRTLIN.

PRTLIN: Prints out a one line summary of each iteration.

QPRINT: Prints out auxiliary quantities when requested and at end of run.

Iterating Subroutines

RELAX: Controlling subroutine for overrelaxation calculation.

LOADCM: Interpolates curvilinear mesh coefficients.

NORM: Normalizes final values to specified stored energy.

SPECIN: Interpolates field values and contributions to numerator and denominator of eigenvalue calculations, from curvilinear meshes to square mesh points with incomplete stars.

SPECFL: Unpacks forward and backward interpolation coefficients previously stored by SETSPC and SETCOF.

UNPACK: Unpacking subroutine

ITERAT: Controls indexing, etc. of iteration processes, for traversal of both square and curvilinear meshes.

ALG: Square mesh algorisms

ALGC: Curvilinear mesh algorisms

RFCNST: Calculates R-1/4R for ALG.

FASET: Saves two just previous rows of old field values for calculation of eigenvalue.

ALFCAL: Calculates the optimum value of alpha the over-relaxation factor.

Interpolating Subroutines

INTERP: Interpolates field values from square mesh to outer row and ends of curvilinear meshes.

Output Subroutines

OUTPUT: Controlling output subroutine

QCALC: Calculates the auxiliary quantities at the end of a run or when demanded.

EZCALC: Performs various axial electric field integrations requested by QCALC.

Appendix C

BCDSUM - Mesh Tape Listing Program

This program is designed to print the eigenfunction, etc. contained in a consecutively numbered sequence of meshes from a MESSYMESH mesh tape. The program requires the tape to be mounted on logical unit 20. Data is input through a FORTRAN (I3,E15,7) format.

Data

Column 3	Column 18	
1	10057.0	mesh run number
2	0.5	scale factor
3	0.0	Type of drift tube 0 = cylindrical >0 = ellipsoidal <0 = odd shape
4	0.0	E_0(MV/m). Average axial electrical field to which output is to be scaled. If 0, E_0 = 1.0 MV/m
5	0.0	flag to not calculate maximum electric field and location on drift tube if non-zero.
9	0.0	various debug prints from subroutine
10	0.0	which computes maximum electric field

Remarks

1. The meshes printed must be stored in successive numerical order on a mesh input/output tape. Violation of this rule will result in a program stop.

Appendix C
(Cont.)

2. Each data address is self perpetuating in a series of runs, unless changed by the agenda.
3. Input data for successive series of meshes must be separated by one blank card; the data for the last series of meshes must be followed by two blank cards.

```
MURA LINAC CAVITY CALCULATIONS                                   RUN NO. 10057

CYLINDRICAL DRIFT TUBE, WITH HOLE    (DIMENSIONS IN CENTIMETERS)
  L =  84.285           SL = 44.56          A = 4.0000          A/L = .0475
  D =  84.000           SD = 16.00        PHC = 1.0000          G/L = .4713
  G =  39.724           RC =  5.00

MESH DIMENSION                               .5000              CM/MESH UNIT

FREQUENCY = 201.249 MC/SEC.      BETA =  .5658       ENERGY = 199.649 MEV

NORMALIZATION FACTOR            FACTOR =  589.05

AVERAGE AXIAL EFIELD             E0 = 1.000    MV/METER    GAUSS LAW
                                 E0 = 1.019    MV/METER    LINE INTEGRAL

STORED ENERGY                     W =     .8843        JOULES

VOLUME OF CELL                    V =     .4592        CUBIC METERS

FREQUENCY PERTURBATION
DUE TO DRIFT TUBE SUPPORT       FFP =  .0005951        *RSTEM**2

POWER DISSIPATION                    WATTS             WATTS/SQUARE METER
  TO OUTER WALLS               PW1 =    6980.38        PW1 =    3138.34
  TO END PLATE                 PW2 =    1397.46        PW2 =    2521.68
  TO DRIFT TUBE PLATE          PW3 =    5580.15        PW3 =   10448.33
  TO DRIFT TUBE                PW4 =   10021.60        PW4 =   34215.54
  TO DRIFT TUBE SUPPORT        PW5 =     635.86        PW5 =   29764.75

QUALITY FACTOR
  FOR LINAC CAVITY               Q1 =  65677.0
  FOR LAB CAVITY                 Q2 =  36070.0

SHUNT IMPEDANCE
  FOR LINAC CAVITY             ZS1 =  49.51             MEGOHMS/METER
  FOR LAB CAVITY               ZS2 =  27.19                   ''

TRANSIT TIME FACTORS             T =  .5540             S =  .7144
  AND DERIVATIVES               TP =  .1230            SP =  .0580
                               TPP =  .0083           SPP =  .0234

PRODUCT ZS1*T**2               ZTT = 15.195             MEGOHMS/METER

PEAK ELECTRIC FIELD           EMAX =  5.675            MILLION VOLTS/METER

PEAK FIELD LOCATION                3.01CM. FROM AXIS OF DRIFT TUBE
                                   0.00CM. FROM END OF DRIFT TUBE

EC= 1.3664 MV/M       HT= 172856.62 AMPS       F1= 492.630 AMPS
                      HC=  1114.435 AMPS       F2= 624.781 AMPS
```